1. Сутність комп'ютерної дискретної математики.

Це комплекс математичних дисциплін, який виник десь в середині минулого століття, а саме тоді, коли почали розвиватися комп'ютерні системи.

Чому?

Тому, що математика – математичний аналіз – він працює з неперервними об'єктами, тобто з тим, що можна ділити, ділити, ділити…, а воно ще ділиться й ділиться. Дискретна математика працює з об'єктами, які можна перерахувати, тобто дискретна – конкретна кількість. І комп'ютерні системи саме так і працюють. Тобто все, що виконується в комп'ютерних системах (його може бути дуже багато, але воно скінченне або зліченне) можна певним описати певною алгоритмічною мовою.

Різні вчені, різні дослідники включають в курс КДМ різні напрями. З того, що включається в КДМ ми в нашому курсі будемо досліджувати першу чергу:

1) теорію множин та теорію відношень (абетка математики) – це ті літери, без яких ви нічого розуміти не будете; це буде перший цикл. Завершимо ми його основами комбінаторики.

2) Потім ми будемо вивчати теорію графів. Графи – це найпотужніший засіб візуалізації процесів і там є певна математична теорія, яка дозволяє все що можна обчислювати, знаходити, доводити. І ми це будемо досліджувати.

3) І завершимо ми теорією автоматів та формальних граматик. По суті це те, як працюють будь-які, в тому числі комп'ютерні програми, як будується програмний код.

2. Теорія множин

Що таке множина?

Георг Кантор (нім. математик, який винайшов цю теорію) запропонував інтуїтивне визначення множини.

<u>Множина</u> (англ. set) – це певна сукупність об'єктів, які ми можемо розрізнити між собою, які не повторюються і які об'єднані в єдине ціле нашою інтуїцією за якоюсь певною ознакою.

Скажімо множина студентів потоку ЖДТУ – це множина. Ось сидять об'єкти. Я вас можу розрізнити поки-що. Ви не повторюєтеся… тут близнят не має, здається. І ви об'єднані в певне ціле, бо ви потік. Множина парт в цій залі – теж зрозуміло. Множина студентів і парт у цій залі. Знову ж таки я можу відрізнити студента від парти, парту від парти, студента від студента.

Тобто ми можемо об'єднувати множини. Фактично все, що завгодно. І тому це визначення підкупає своєю простотою. Всім зрозуміло що таке множина. Ось це взяли – і це множина.

Але ми називаємо це інтуїтивним визначенням, то воно не є формальним, воно не є математичним. І наприкінці лекції ми побачимо, що ось таке визначення призводить до парадоксів, тобто до об'єктів, які фізично не можуть існувати.

Але поки-що, якщо у нас уже є такий математичний об'єкт, – треба навчитися з ним працювати.

В першу чергу треба навчитися його зображати, щоб всі розуміли про що йдеться.

3. Способи подання (представлення) множин.

Їх є 2,5.

1) **Явний**. Ми беремо і перелічуємо всі об'єкти, які входять до даної множини. Наприклад, множина A={a, b, c, …, z} – множина всіх літер латинського алфавіту. Множина B={0, 1} – це множина значень, які можуть приймати певні біти у комп'ютерних системах. Множина C={ялинка, їжачок}. Вона складається з ялинки та їжачка. А множина D={0, 1, ялинка} з нуля, одиниці та ялинки. Тобто все, що завгодно. Ми використовуємо певні службові символи. Множина позначається фігурними дужками, об'єкти перелічуються через кому. При цьому порядок об'єктів у множині не важливий. Тобто, якщо у мене є множина $B_2 = \{1,0\}$, що складається з 1 та 0, то це всерівно, що множина B, яка складається з 0 та 1: $B_2 = \{1,0\} = \{0,1\}$.

Є дві складнощі в такому представлені.

Перша очевидна. Там використано «…» – якщо об'єктів забагато, то їх перелічення буде займати дуже багато часу та місця, більш того – у мене може бути нескінченна множина, тобто об'єктів буде настільки багато, що їх не можна перелічити за розумний час.

І ще одна така синтаксична неув'язка. Якщо хочете включити до множини об'єкт «,» або об'єкт «{». Це рідкісні випадки, але є такі. Тоді використовуйте «» до окремого службового слова. У нас такого не буде.

Але що робити з великими множинами?

2) **Неявний спосіб**.

Ми об'єднуємо у множину всі об'єкти, що задовольняють певні властивості.

Нехай $P(x)$ – це певна властивість (науковою мовою – предикат).

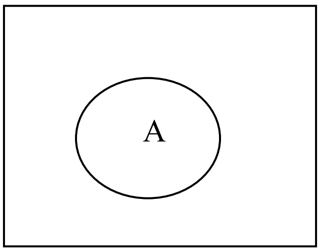
Істина: якщо ця властивість виконується – 1, та неістина (хиба), якщо ця властивість не виконується –0. Тоді можемо задати множину як всі елементи *x*, для яких виконується задана властивість: $X = \{x : P(x)\}$ *або* $X = \{x \mid P(x)\}$.

<u>Наприклад</u>.

$E = \{x \mid x - \textit{натуральне}, x - \textit{парне}\}$ – множина всіх парних натуральних чисел задається за допомогою такого предиката, який ми описуємо словом. Можемо описувати словами, можемо більш формально.

3) Третій **спосіб графічний**.

Це не спосіб представлення множини. <u>Це спосіб її візуалізації</u>. Називається **діаграма Ойлера-Венна**.



Це множина A. Це такі кола, можуть бути не кола. Діаграмою не можна подати множину. Ми не будемо знати, які там є об'єкти, але можна дуже зручно візуалізувати операції над множинами.

4. Стандартизовані множини

У нас є множини, які певним чином стандартизовані.

Для них є спеціальні позначення:

Ø – порожня множина – це множина, яка не містить жодного елементу.

U – множина універсум – це множина, яка містить всі об'єкти. Далі ми покажемо, що універсаму не існує. Але поки що ми його введемо.

N (з товстою ніжкою) – множина натуральних чисел: $N = \{1,2,3,...\}$.

Підкреслимо, що в нашому курсі 0 (нуль) не буде вважатися натуральним числом. Тут є цікава ситуація: у Франції, Бразилії 0 вважається натуральним числом; у інших країнах 0 не вважається натуральним числом. Але іноді його зручно вважати натуральним числом, долучати до множини натуральних чисел. Але у викладі нашого курсу зручно вважати 0 не натуральним числом.

Але якщо 0 нам все таки знадобиться, – ми введемо множину

N_0 – цілих невід'ємних чисел: $N_0 = \{0,1,2,3,...\}$(natural – натуральне).

Насправді в природі існують лише натуральні числа, а все інше – це вигадка людського мозку.

Наприклад.

Z – множина цілих чисел: $Z = \{0,1,2,...\}$ Чому Z? (Георг Кантор був німцем: і от Zahlen (цалєн) – це числа). Для нього цілі числа були «паскудними».

Q – раціональні числа: $Q = \left\{ \frac{m}{n} \mid m \in Z, n \in N \right\}$. І, надіюсь, тепер зрозуміло, чому я не хочу, щоб 0 був N-числом.

Чому Q? Від лат. quotient – частка.

Ще дві множини залишилося.

R – множина дійсних чисел. Тут я її вже не можу ні описати, ні перелічити. R – тому, що real – «дійсне».

І остання множина – це комплексні числа – C (komplex – склади).

Тобто в нас уже є певне розуміння, що таке множина; у нас уже є певні множини, з якими ми довго працювали, але не знали, що це є множини і вони певним чином позначаються.

Що ми можемо з ними зробити?

По перше, перевірити чи належить елемент множині, чи ні: $a \in A$ (українською мовою це досить легко читати: «елемент а є елементом множини А»). «$\in$» – це, дійсно, латинська літера «є» певним чином трансформована.

5. Синтаксис.

Ви як майбутні програмісти повинні чітко дотримуватися синтаксису. Якщо у вас є такий символ ($\in$), то праворуч від нього обов'язково повинна стояти множина, а ліворуч – все, що завгодно і в нашому розумінні може бути елементом цієї множини (може розглядатися як елемент цієї множини). Тобто може стояти інша множина, наприклад, числа, букви, об'єкти, матриці, поліноми, парти, студенти, зірки у Всесвіті, – все, що завгодно.

Якщо я пишу: $a \notin A$, то об'єкт а не є елементом множини А. Він не належить цій множині. Тут все просто, здається. Так?

Тобто ваша задача зараз – навчитися правильно читати і правильно писати.

Включення – це відношення між множинами. $A \subseteq B$ – «множина А включається в множину В» або «множина А є підмножиною множини В». Якщо писати це відношення словами, то це означає, що всі елементи множини А одночасно є елементами множини В.

Оскільки у нас математика, – ми стверджуємо лише це і більше нічого. Тобто ми не знаємо чи є у В ще якісь елементи, окрім тих, що є в А. Чи є в А взагалі якісь елементи.

Наприклад: $\varnothing \subseteq B$ – **порожня множина є підмножиною довільної множини**. Чому? Тому, що всі елементи цієї множини (Ø) повинні належати цій множині (В). А які тут (Ø) є елементи? Ніякі. Тобто чи можу я порушити цю умову? Чи можу я навести елемент звідси (Ø), щоб належав (В)? Не можу. Тобто це твердження виконується.

І ось цей принцип, коли для того, щоб порушити певне твердження, ви повинні навести контр-приклад. Якщо не можете навести контр-приклад, то ви не можете порушити. Він у нас буде використовуватися дуже часто.

Тепер оце словесне твердження: всі елементи множини А одночасно є елементами множини В. Тепер я його напишу, як його математики пишуть: $(A \subseteq B) \Leftrightarrow (\forall a : a \in A \Rightarrow a \in B)$, тобто множина А є підмножиною множини В тоді та тільки тоді, коли для будь-якого елементу а з того, що а належить множині А випливає, що а належить множині В. Чи всім зрозумілий цей запис?

$\Leftrightarrow$ – логічна еквівалентність: тобто ось це твердження $(A \subseteq B)$ виконується тоді і тільки тоді, коли виконується $(\forall a : a \in A \Rightarrow a \in B)$.

$\forall$ – квантор загальності. Він читається як «будь-який», «довільний» або «всі». І взагалі це анг. слово $\forall$ny . Насправді не анг., а французьке. Але англійське я не пам'ятаю, тому пишу від себе англійське. Це літера А, яку перевернули.

Є ще квантор $\exists$ – існування, який читається як «існує» (може з уточненням «існує деякий»). І це від англ. слова exist – «існує».

Тобто те, що іде після «:» у $(\forall a : a \in A \Rightarrow a \in B)$ виконується для будь-якого елементу a. Що виконується? Логічний наслідок $(\Rightarrow)$ якщо твердження тут $(a \in A)$ є коректним, то твердження тут $(a \in B)$ також повинне бути коректним. Якщо твердження тут $(a \in A)$ не є коректним, то нам все рівно. Це дуже важливо насправді, тому що $\varnothing \subseteq B$. Щоб зробити це твердження $(a \in A)$ некоректним треба підібрати такий елемент a, для якого ось це твердження $(a \in A)$ буде невірним, а ось це твердження $(a \in B)$ буде вірним. Але ми не можемо підібрати таких елементів a, тому що таких елементів в порожній множині не існує. Тому я не можу порушити цей логічний наслідок. Для всіх елементів a, ось це твердження $(a \in A)$, якщо тут підставити Ø воно буде некоректним. Логічний наслідок не порушується. Типовою помилкою початківців є те, що вони бачать, що ліва частина логічного наслідку є некоректна, – і все: сам логічний наслідок є некоректним. Ні. Наслідок перевіряє на це.

Строге включення.

$A \subset B$ – «всі елементи множини A є елементами множини B, але в множині B є ще елементи, яких немає в множині A».

Математично: $(A \subset B) \Leftrightarrow (A \subseteq B) \overset{\wedge}{\&} (\exists b \in B : b \notin A)$

Ми кажемо, що «множина A є строгою підмножиною множини B тоді та тільки тоді, коли A є підмножиною множини B і одночасно з цим (& – амперсанд; або можна писати – $\wedge$ (перевернуту літеру v); це одне і теж саме) існує такий елемент b множини B, для якого виконується, що b не належить множині A».

У нас новий символ – & ($\wedge$) – це логічне «та». Ми його ставимо, коли обидва твердження повинні бути істинними одночасно.

Альтернативою є $\vee$ – логічне «або» (це альтернативна істинність). Або одне, або інше, або обидва одночасно.

Чи можу я стверджувати, що $\varnothing \subset B$ довільної множини B? Для довільної множини не можу. Чому? Якщо B є порожньою множиною, то це твердження не є коректним. Чому? Тому, що не виконується $(\exists b \in B)$, тобто тут не буде існувати жодного b.

Якщо B є не порожньою множиною, тоді **так**. Тобто ми зможемо підібрати такий елемент, який не буде належати порожній множині, оскільки цій множині не належить жоден елемент.

6. Що таке рівність двох множин?

Ми кажемо, що **дві множини рівні $A = B$, якщо вони складаються з однакових елементів**. Це інтуїтивно зрозуміло. Але з'являються вже певні складності.

Ми можемо це перевірити, коли наші множини подані явно. Тобто ми бачимо всі їх елементи. Але якщо множини подані неявно, то треба перевіряти предикати. А це вже складно. Тому ось це словесне визначення ми будемо замінювати на формальне визначення, з яким будемо працювати. Будемо казати що дві множини рівні тоді та тільки тоді, коли множина A є підмножиною множини B і одночасно всі елементи B належать множині A: $(A = B) \Leftrightarrow (A \subseteq B) \& (\mathrm{B} \subseteq \mathrm{A})$.

Тепер нам треба навчитися робити операції над множинами. У нас таких операцій існує декілька.

7. Операції над множинами.

1) **об'єднання** (англ. union): **операцією об'єднання двох множин ми називаємо множину, яка складається з елементів першої та другої множини.** От зараз я підкреслюю, що це визначення, яке я сказала, я його, мабуть, напишу словами, щоб ви бачили форму, якою йдуть визначення в дискретній математиці. «**Множина C називається об'єднанням множин A та B, якщо це множина, що складається з**

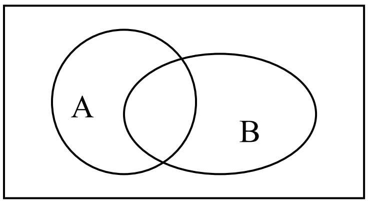
елементів, які належать множині А або належать множині В»: $C = A \cup B = \{x : (x \in A) \vee (x \in B)\}$.

Для чого витратила купу часу зараз? Бачите як будується визначення:

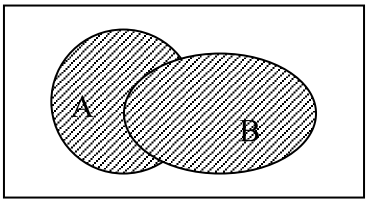
по-перше: що ми визначаємо? Ми визначаємо об’єднання двох множин;

якщо мені потрібно оперувати у визначенні ось цими множинами (А та В) – я їх іменую. Тобто я могла б сформулювати: об’єднанням двох множин – це…; а можу формулювати: об’єднання множин А та В… і можу *директивно звертатись* до множини А та В окремо. Так «називається», тобто тут ми переходимо до визначення. В першу чергу ми вказуємо – який математичний об’єкт визначається. Тобто «об’єднання множин – це **множина**». Не число, не літера… це множина. Далі ми кажемо як визначається ця множина, тобто яким чином вона будується. Фактично ми описуємо ось цей предикат ($(x \in A) \vee (x \in B)$) словом. Я від вас вимагатиму формулювання саме в такому вигляді.

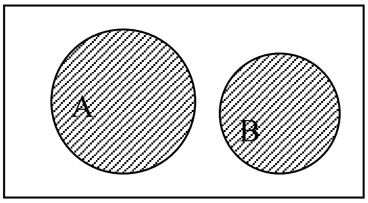
Зрозуміти що таке об’єднання нам допомагає графічне представлення.



Ось у нас є множина А, ось у нас є множина В. Що є об’єднанням цих двох множин? Множина С.



C



C

Якщо в об’єднанні є елементи, які належать і А і В, – в об’єднанні вони рахуються 1 раз. Тому, що елементи всередині множини не повторюються.